

## Tema 2. Spanning Tree Protocol

Administración de Redes

## 1- Contextualización

Cuando conectas más de un cable o más switches entre sí, estás creando caminos alternativos por si uno falla.

Esto mejora la disponibilidad: si algo se rompe, hay otra ruta.

- La redundancia de capa 2 mejora la disponibilidad de la red implementando rutas de red alternas mediante la agregación de dispositivos de red y cableado
- El inconveniente es que si hay más de un camino para enviar una trama en capas 1 y 2 (redundancia) se produce un bucle infinito como resultado

los switches reciben tramas y las envían al destino según la tabla de direcciones MAC. Si no saben a donde va se envía por todos los puertos excepto por donde vino

- 1 – Operativa de envío de tramas en switches Ethernet
- 2 – Las tramas Ethernet no tienen atributo TTL para se eliminadas de la red
- 3 – Las tramas en la red conmutada se continuarán enviando entre switches hasta el infinito (y más allá)
- 4 – Esta propagación continua entre switches provocará inestabilidad en las tablas CAM de los switches

\*2-A diferencia de los paquetes IP (que tienen un TTL: Time To Live), las tramas Ethernet no caducan.

Pueden circular por la red infinitamente si no se controlan.

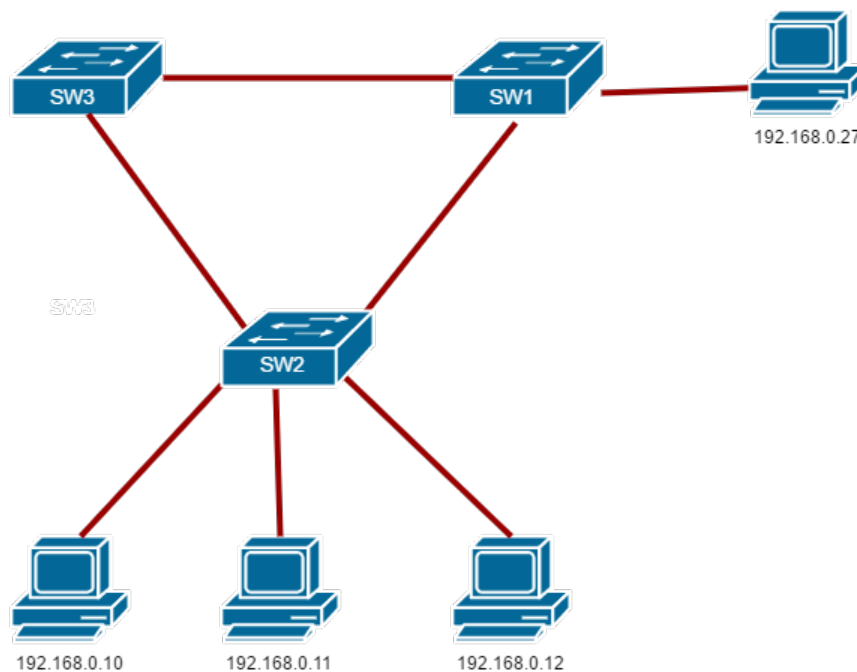
\*3-Sin STP, una trama puede rebotar entre switches sin fin, ocupando ancho de banda y colapsando la red.

\*4-Los switches usan una tabla CAM para saber por qué puerto está cada dirección MAC. •2

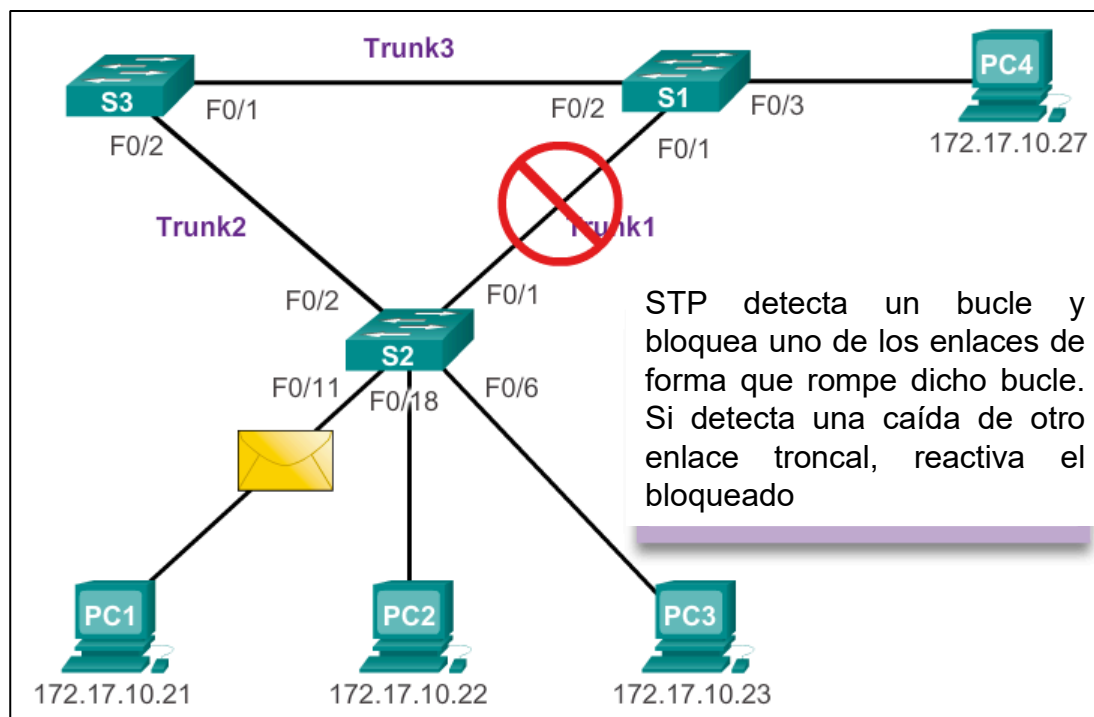
Si la misma trama llega por muchos puertos diferentes (por el bucle), el switch se confunde y borra o cambia la información erróneamente.

Resultado: la red deja de funcionar bien.

## Contextualización



## Contextualización



## 2-Objetivos de STP

- Evitar las tormentas de *broadcast* : STP bloquea caminos redundantes para que los mensajes broadcast no circulen sin fin en la red.
- Evitar la inestabilidad de las tablas de envío de los switches : Sin STP, los switches podrían confundirse con información contradictoria y cambiar constantemente su tabla MAC, causando problemas.
- Evitar la duplicidad de tramas *unicast*

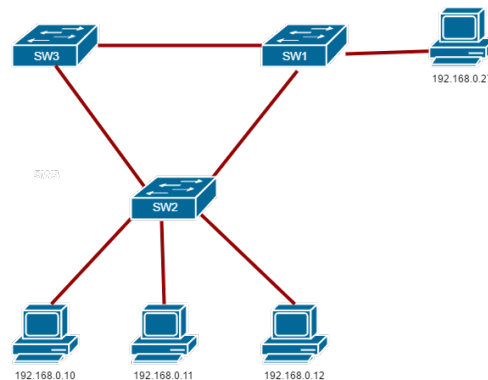
STP evita que la misma trama unicast (una trama que se envía de un dispositivo a otro específico) se envíe por varios caminos, evitando confusión y sobrecarga en la red.

## Objetivos de STP: Tormentas de Broadcast

- Los switches envían las tramas de broadcast por todos los puertos excepto por el que la han recibido
- Si se crea un bucle de capa 2, las tramas de broadcast se enviarán de forma continua, sin fin. Esto provocará una “tormenta de broadcast”
- Debido a esto, no habrá ancho de banda disponible para más tráfico y la red quedará bloqueada para enviar tráfico de datos. Se produce por lo tanto una denegación de servicios
- Una tormenta de broadcast es inevitable en una red de capa 2 con bucles
- Hay gran cantidad de tráfico de broadcast que es necesario para el funcionamiento de otros protocolos (ARP, DHCP, ...)
  - Por lo tanto, este tipo de tráfico consumiría todo el ancho de banda y provocará que la red falle

## Objetivos de STP: Inconsistencia en las tablas de envío

- Sin STP, las tramas de difusión se retransmiten de forma continua por los bucles formados en la red, consumiendo el ancho de banda disponible
  - Dichas tramas tienen la dirección MAC de destino FF-FF-FF-FF-FF-FF pero la dirección MAC de origen es la dirección de la tarjeta de red del dispositivo que ha generado la petición
- En el ejemplo, supongamos que 192.168.0.10 necesita averiguar la MAC de 192.168.0.12. Para ello envía una petición ARP, con su propia dirección MAC como origen y la dirección de *broadcast* como destino.
  - La trama se reenvía por todos los puertos, menos por el origen
  - Por lo tanto, se envía, entre otros, a SW1 y SW3
  - Estas dos tramas circulan de forma continua por el bucle, haciendo que la dirección MAC del equipo 192.168.0.10 vaya cambiando de puerto en los tres switches



## Objetivos de STP: Tramas Unicast Duplicadas

- Las tramas *unicast* que se envían a través de una red con bucles puede provocar que lleguen tramas duplicadas al dispositivo destino.
- La mayor parte de protocolos de capa superior no están diseñados para reconocer o enfrentarse con transmisiones duplicadas. En general, los protocolos que hacen uso de mecanismos de números de secuencia podrían asumir que la transmisión a fallado y descartarán los paquetes recibidos. Los protocolos de capa superior, que no utilizan estos mecanismos, no podrán detectar este problema
- Los protocolos de capa 2, como Ethernet, no tienen mecanismos para reconocer o eliminar tramas que se envían de forma continua a través de un bucle

-Protocolos que sí manejan tramas repetidas -> NO PASARÍA NADA CON LAS TRAMAS UNICAST DUPLICADAS

Usan números de secuencia (como TCP) y pueden descartar duplicados.

-Protocolos que no manejan bien tramas repetidas ->CAUSARÍAN ERRORES LAS TRAMAS UNICAST DUPLICADAS

Capa 2 (Ethernet) y otros sin control, no detectan duplicados y causan errores.



## 2- El Algoritmo Spanning-Tree: Introducción

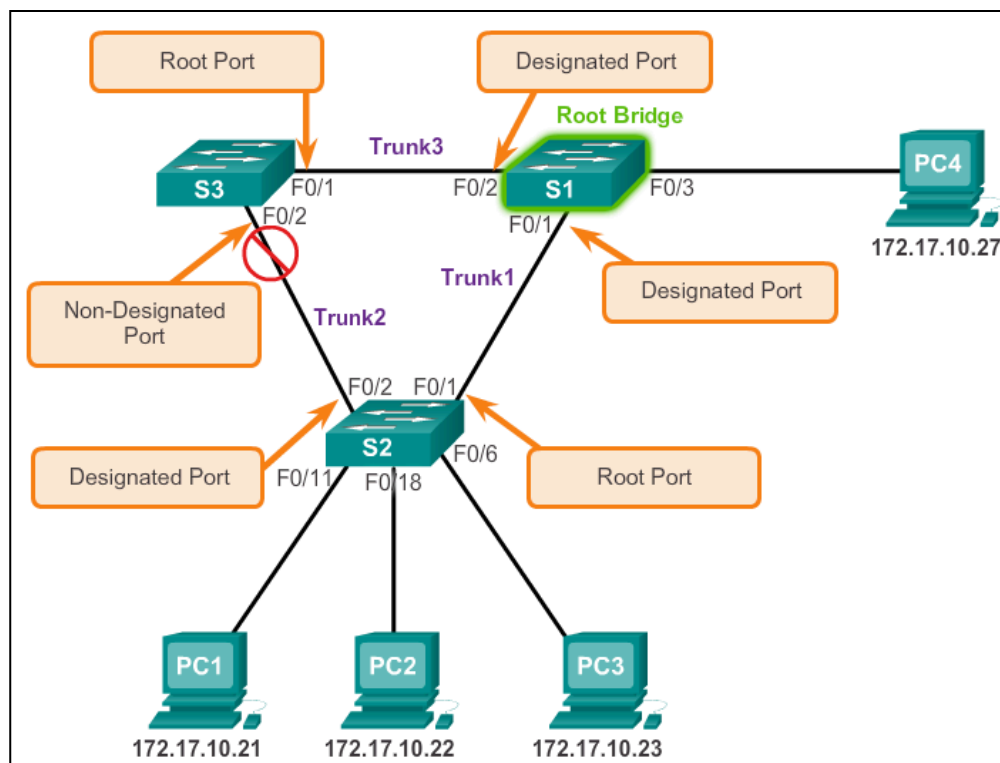
- STP garantiza que solamente hay **un camino lógico entre todos los destinos** de una red conmutada, **bloqueando intencionadamente los caminos redundantes** que podrían provocar un bucle
- Se considera que un puerto está bloqueado cuando los **datos de usuario** no se envían ni se reciben por dicho puerto.
  - Esto no incluye las tramas utilizadas por el propio protocolo STP, denominadas “**Bridge Protocol Data Unit**” (BPDU), que se utilizan para detectar y evitar los bucles
- Pueden existir rutas físicas redundantes en una red conmutada, pero estas rutas estarán desactivadas para evitar bucles de conmutación -> las rutas existen físicamente igual, pero STP las desactiva lógicamente para evitar los bucles
- Si la ruta se vuelve necesaria para compensar un fallo en otro enlace o en un switch, STP recalcula los caminos y desbloquea los puertos que sean necesarios para permitir que la ruta redundante que estaba bloqueada pase a estar activa.

Basicamente, si un enlace se cae STP vuelve a recalcular el camino pudiendo desbloquear los puertos que antes había bloqueado y bloqueando otros ahora como aquellos que fallaron.

## El Algoritmo Spanning-Tree: Bridge Designado

- STP utiliza el **algoritmo Spanning Tree (STA)** para determinar los puertos de switch de la red que deben bloquearse y así evitar que se generen bucles.
  - **STA elige un único switch como puente raíz** y lo utiliza como punto de referencia para todos los cálculos de rutas.
- Todos los switches que comparten STP intercambian **tramas de BPDU** para determinar el switch que posee el **menor ID de puente (BID)** en la red.
  - **El switch con el menor BID** se transforma en el **puente raíz** de forma automática según los cálculos del STA.
  - **Cada BPDU contiene un BID** que identifica al switch que envió la BPDU.
  - El **BID** contiene un **valor de prioridad**, la **dirección MAC** del switch emisor y una ID de sistema extendido optativa
- De manera predeterminada, las tramas de BPDU se envían cada 2 segundos después de iniciar el switch

## El Algoritmo Spanning-Tree: Roles de los Puertos



# El Algoritmo Spanning-Tree: Bridge Designado

- **Prioridad del puente**

- La prioridad del puente es un valor que puede personalizarse y puede utilizarse para ejercer influencia sobre el switch que debe convertirse en el puente raíz.
- **El switch con la menor prioridad, es decir, el menor BID, se transforma en el puente raíz**
- El **valor predeterminado** de la prioridad para todos los switches de Cisco es **32.768**.
- El rango de prioridad oscila entre 0 y 61440; por lo tanto, **0 es la prioridad más alta**.
- Los valores de prioridad de puente sólo pueden ser múltiplos de 4096. (0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344 y 61440)

# El Algoritmo Spanning-Tree: Bridge Designado

- **Cambiar la Prioridad**

- **Método 1:**

- Para asegurar que el switch posea el menor valor de prioridad de puente, se puede utilizar el comando en modo de configuración global

```
switch(config)# spanning-tree vlan vlan-id root primary
```

- La prioridad del switch se establece en el **valor predefinido 24.576** o en el siguiente valor de incremento de 4096 por debajo de la menor prioridad de puente detectada en la red.
    - Si desea contar con un puente raíz alternativo, utilice el comando en modo de configuración global.

```
switch(config)# spanning-tree vlan vlan-id root secondary
```

- Este comando establece la prioridad para el switch al **valor preferido 28.672**

# El Algoritmo Spanning-Tree: Bridge Designado

- **Cambiar la Prioridad**

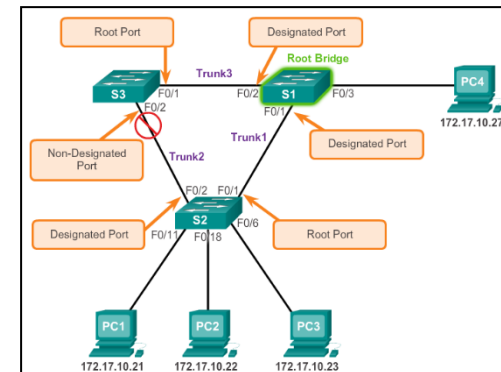
- **Método 2:**

- Otro método para configurar el valor de prioridad de puente es mediante el comando en modo de configuración global  
**Switch(config)# spanning-tree <vlan vlan-id> priority <valor>**
    - Este comando proporciona más control granular sobre el valor de prioridad de puente
    - Después de determinar el puente raíz, STA calcula la ruta más corta hacia el mismo
    - Todos los switches utilizan el STA para determinar los puertos que deben bloquearse.

# El Algoritmo Spanning-Tree: Roles de los Puertos

- **Puerto raíz**

- El puerto raíz existe en los puentes que no son raíz
- Es el puerto de switch con el mejor camino hacia el puente raíz.



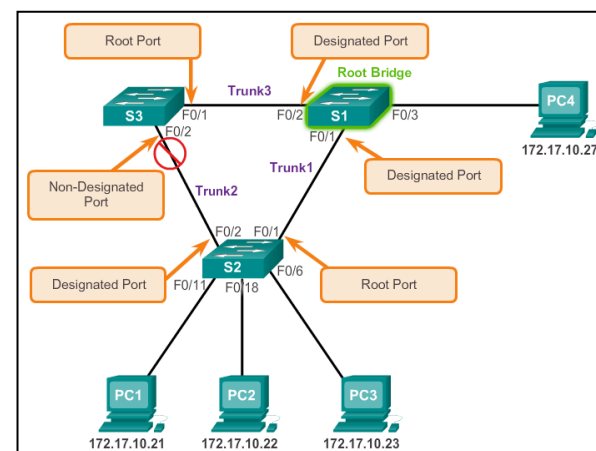
- **Puerto designado**

- El puerto designado existe en los puentes raíz y en los que no son raíz
  - **¿Para los switches raíz, todos los puertos de switch son designados?**
  - Para los puentes que no son raíz, un puerto designado es aquel que recibe y envía tramas a través del puente raíz según sea necesario.
  - Sólo se permite **un puerto designado por segmento** (el más próximo al switch raíz)

# El Algoritmo Spanning-Tree: Roles de los Puertos

- **Puerto no designado**

- **El puerto no designado es aquel puerto de switch que está bloqueado**
- No envía tramas de datos ni aprende la tabla de direcciones MAC con direcciones de origen.

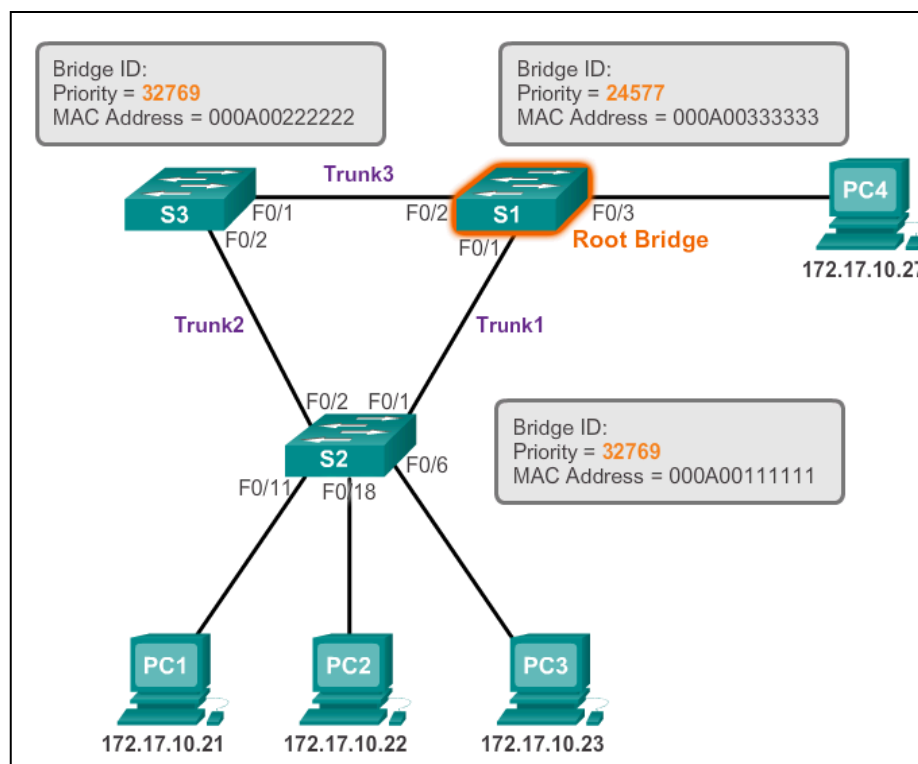


- **Puerto deshabilitado**

- El puerto deshabilitado es un puerto de switch que está administrativamente desconectado.
- Un puerto deshabilitado no funciona en el proceso de *spanning-tree*.



## El Algoritmo Spanning-Tree: Bridge Designado



## El Algoritmo Spanning-Tree: Coste de la Ruta

- STA considera los costes tanto de la ruta como del puerto cuando determina la ruta que debe permanecer desbloqueada.
  - **Los costes de la ruta se calculan mediante los valores de coste de puerto asociados con las velocidades de los puertos.**
- La suma de los valores de coste de puerto determina el coste de ruta total al puente raíz.
- Los costes de los puertos predeterminados se definen por la velocidad a la que funcionan los mismos.

Link Speed	Cost (Revised IEEE Specification)	Cost (Previous IEEE Specification)
10 Gb/s	2	1
1 Gb/s	4	1
100 Mb/s	19	10
10 Mb/s	100	100

## El Algoritmo Spanning-Tree: Coste de la Ruta

- Pese a que los puertos de switch cuentan con un coste de puerto predeterminado asociado a los mismos, tal coste puede configurarse.
- El rango de valores puede oscilar entre 1 y 200.000.000

### Configurar costo del puerto

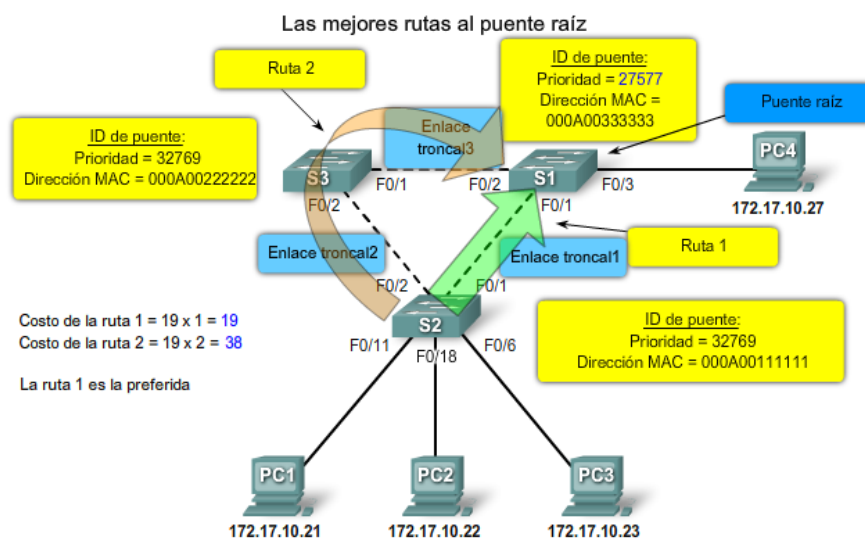
```
S2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)#interface f0/1
S2(config-if)#spanning-tree cost 25
S2(config-if)#end
S2#
```

### Restablecer costo del puerto

```
S2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)#interface f0/1
S2(config-if)#no spanning-tree cost
S2(config-if)#end
S2#
```

## El Algoritmo Spanning-Tree: Coste de la Ruta

- El coste de la ruta es la suma de todos los coseos de puertos que atraviesan la ruta hacia el puente raíz.
- La ruta con el menor coste de ruta se convierte en la ruta preferida y todas las demás rutas redundantes se bloquean.



## El Algoritmo Spanning-Tree: Coste de la Ruta

- Para verificar el coste de puerto y de ruta hacia el puente raíz, utiliza el comando del modo EXEC privilegiado **show spanning-tree**
- El campo **cost** del resultado es el coste de ruta total hacia el puente raíz.
  - Este valor cambia en función de la cantidad de puertos de switch necesarios para llegar al puente raíz.

Las mejores rutas al puente raíz

```
S2#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority 27577
            Address  000A.0033.3333
            Cost    19
            Port    1
            Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

  Bridge ID  Priority 32769 (priority 32768 sys-id-ext 1)
            Address  000A.0011.1111
            Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
            Aging Time 300

Interface Role Sts Cost Prio.Nbr Type
-----
F0/1      Root FWD 19 128.1 Edge P2p
F0/2      Desg FWD 19 128.2 Edge P2p
```

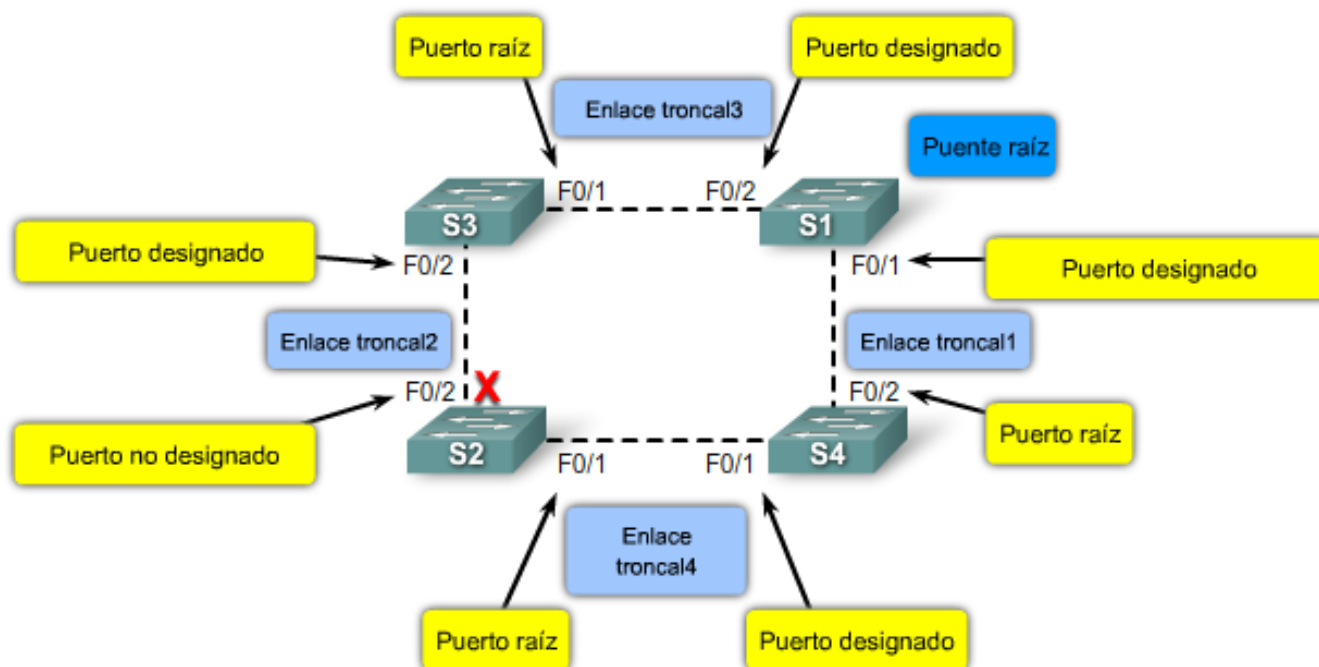
## El Algoritmo Spanning-Tree: Roles de los Puertos

- Cuando existen dos puertos de switch con el mismo coste de ruta hacia el puente raíz y ambos son los de menor coste de ruta en el switch, este último debe determinar cuál de los dos es el puerto raíz.
- **El switch utiliza el valor de prioridad de puerto personalizable o el menor ID de puerto si ambos valores de prioridad de puerto coinciden (en base a las BPDUs recibidas)**
- El ID de puerto es el ID de interfaz del puerto de switch
  - El ID de puerto está adjunto a la prioridad del puerto.
    - El puerto de switch F0/1 posee un valor de prioridad de puerto predeterminado de **128.1**, donde 128 es el valor de prioridad de puerto y .1 el ID de puerto.
    - El puerto de switch F0/2 posee un valor de prioridad de puerto de **128.2** de manera predeterminada.
- Se puede configurar el valor de prioridad del puerto a través del comando en modo de configuración de interfaz.

```
switch(config)# spanning-tree port-priority "valor"
```

Valores válidos: 0 a 240, múltiplos de 16

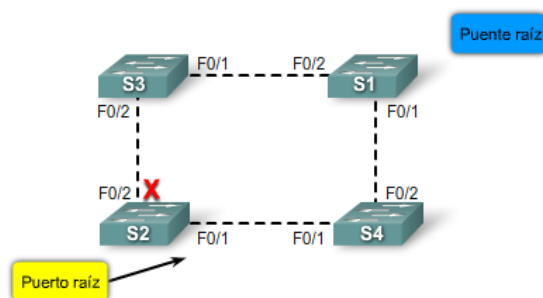
## El Algoritmo Spanning-Tree: Roles de los Puertos



## El Algoritmo Spanning-Tree: Roles de los Puertos

- Los valores de prioridad de puerto oscilan entre 0 y 240, en incrementos de 16.
- El valor de prioridad de puerto predeterminado es 128.
- Al igual que con la prioridad de puente, **los valores de prioridad de puerto menores proporcionan al puerto una mayor prioridad.**

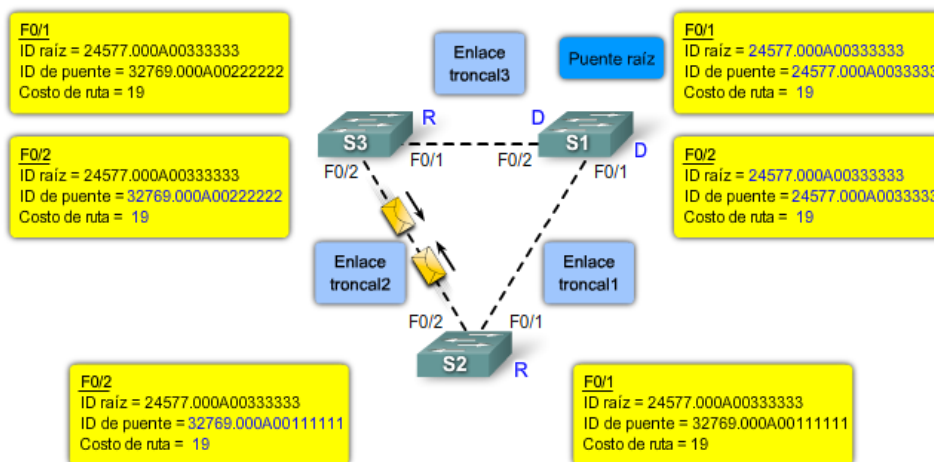
```
S2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)#interface f0/1
S2(config-if)#spanning-tree port-priority 112
S2(config-if)#end
S2#
```





## El Algoritmo Spanning-Tree: Roles de los Puertos

- Cuando dos switches están conectados al mismo segmento de LAN y los puertos raíz ya se han definido, los dos switches deben decidir el puerto que debe configurarse como **designado** y el que debe permanecer como **no designado**.



El switch S2 y el switch S3 intercambian tramas BPDUs. El switch 3 identifica que el switch S2 tiene un BID inferior basado en la dirección MAC inferior del switch S2.

## 4-Spanning-Tree: Funcionamiento

- Si un puerto de switch experimenta una transición directa desde el estado de bloqueo al estado de enviar, dicho puerto puede crear temporalmente un bucle de datos si el switch no advierte toda la información de la topología en ese momento.
- Por esta razón, STP introduce **cinco estados de puertos**
  1. **Bloquear**: el puerto es un puerto no designado y no participa en el envío de tramas. El puerto recibe tramas de BPDU para determinar la ubicación y el ID de raíz del switch del puente raíz y las funciones de puertos que cada uno de éstos debe asumir en la topología final de STP activa.
  2. **Escuchar**: STP determina que el puerto puede participar en el envío de tramas de acuerdo a las tramas de BPDU que el switch ha recibido hasta ahora. En este momento, el puerto de switch no sólo recibe tramas de BPDU, sino que también transmite sus propias tramas de BPDU e informa a los switches adyacentes que éste se prepara para participar en la topología activa.

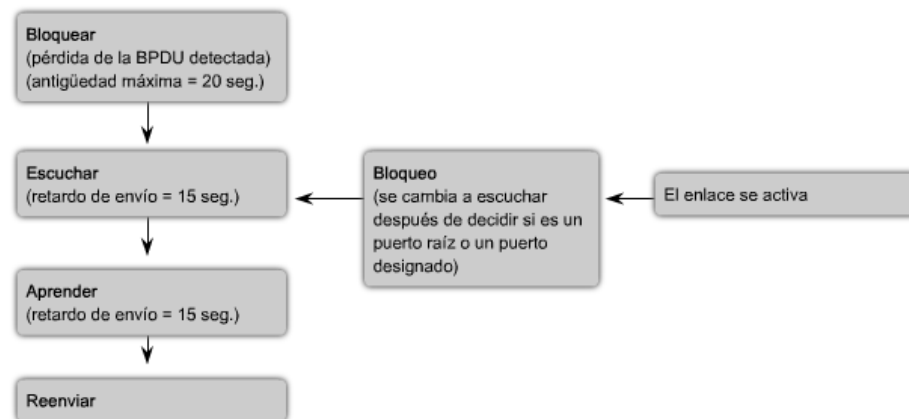
## Spanning-Tree: Funcionamiento

3. **Aprender:** el puerto se prepara para participar en el envío de tramas y comienza a llenar la tabla de direcciones MAC.
4. **Reenviar:** el puerto se considera parte de la topología activa, envía tramas y envía y recibe tramas de BPDU.
5. **Deshabilitado:** el puerto de la Capa 2 no participa en el *spanning tree* y no envía tramas. El estado deshabilitado se establece cuando el puerto de switch se encuentra administrativamente deshabilitado.

Procesos	<i>Blocking</i>	<i>Listening</i>	<i>Learning</i>	<i>Forwarding</i>	<i>Disabled</i>
Recibir y procesar BPDUs	Sí	Sí	Sí	Sí	No
Enviar BPDUs	No	Sí	Sí	Sí	No
Aprender MACs	No	No	Sí	Sí	No
Enviar tráfico de datos	No	No	No	Sí	No

# Spanning-Tree: Funcionamiento

- Todos los puertos de switch de la red atraviesan el estado de bloqueo y los estados transitorios escuchar y aprender al iniciarse.
- Luego los puertos se estabilizan al estado de enviar o de bloqueo.



- Durante un cambio en la topología, el puerto implementa temporalmente los estados escuchar y aprender durante un período de tiempo específico denominado intervalo de retardo de envío.
- Estos valores permiten el tiempo adecuado para la convergencia en la red con un diámetro de switch de valor siete.

## Spanning-Tree: Funcionamiento

- La cantidad de tiempo que un puerto permanece en los distintos estados depende de los temporizadores de BPDU.
- Sólo el switch con función de puente raíz puede enviar información a través del árbol para ajustar los temporizadores.
- Los siguientes temporizadores determinan el rendimiento de STP y los cambios de estado:
  - Hello Time: es el tiempo que transcurre entre los envíos de dos *BPDU*s consecutivas
    - Período entre *BPDU*s
    - Valor por defecto: 2 s – Intervalo válido: 1 a 10 s
  - Forward Delay: es el tiempo que permanece un puerto en los *listening* y *learning*, cuando está progresando hacia el estado *forwarding*
    - Valor por defecto: 15 s – Intervalo válido: 4 a 30 s
  - Maximum Age: es el tiempo que se mantiene la información aprendida mediante una *BPDU* recibida en un puerto del switch. También es el tiempo que un puerto permanece en estado *blocking*, antes de pasar a *listening* si está progresando hacia *forwarding*
    - Valor por defecto: 20 s – Intervalo válido: 6 a 40 s

## Spanning-Tree: Convergencia STP

- Para comprender el proceso de convergencia de forma más profunda, el mismo se ha dividido en tres pasos distintos:
  - **Paso 1.**
    - Elegir un puente raíz
  - **Paso 2.**
    - Elegir los puertos raíz
  - **Paso 3.**
    - Elegir los puertos designados y no designados
- La convergencia es el tiempo que le toma a la red determinar el switch que asumirá la función del puente raíz, atravesar todos los otros estados de puerto y configurar todos los puertos de switch en sus funciones de puertos finales
- El proceso de convergencia demora un tiempo en completarse debido a los distintos temporizadores que se utilizan para coordinar el proceso.

## Spanning-Tree: Convergencia STP

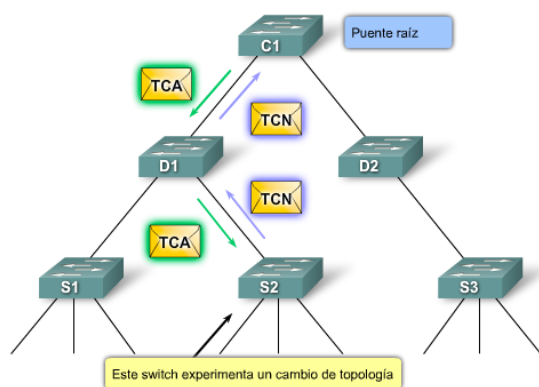
- El proceso de determinar las funciones de los puertos se produce en forma conjunta con la elección del puente raíz y con la designación del puerto raíz.
  - En consecuencia, las funciones de designado y no designado pueden cambiar varias veces durante el proceso de convergencia hasta que se haya determinado el último puente raíz.
- El proceso completo de seleccionar el puente raíz, determinar los puertos raíz y los puertos designados y no designados se lleva a cabo dentro de los 20 segundos que transcurren durante el estado de bloqueo.
- Este tiempo de convergencia se basa en el **hello time** de 2 segundos para las transmisiones de tramas de BPDU y el diámetro de siete switches que admite STP.
- **Maximum age**, de 20 segundos, proporciona el tiempo suficiente para detectar bucles antes de empezar a transmitir:
  - Para un diámetro de siete switches con el **hello time** de 2 segundos

## Spanning-Tree: Notificación de cambio en la topología de STP

- Cuando se detecta un cambio, el switch se lo notifica al switch raíz de *spanning tree*.
- Después, el switch raíz hace difusión de esta información a toda la red
- Para notificar un cambio de topología, un switch comienza a enviar TCNs (*Topology Change Notification*) por su puerto raíz.
  - Una TCN es una BPDU muy simple que no contiene más información que la detección del cambio y se envía periódicamente en base al *hello time*
  - El switch receptor inmediato de dichas TCNs se denomina **switch designado**
    - Realiza el acuse de recibo de la TCN mediante el envío inmediato de una BPDU normal con el bit de acuse de recibo de cambio en la topología (TCA).
  - Este intercambio continúa **hasta llegar al switch raíz**



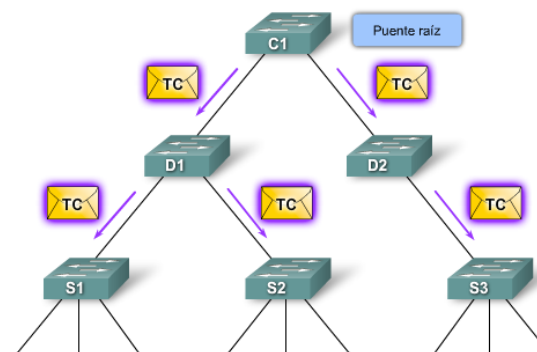
# Spanning-Tree: Notificación de cambio en la topología de STP



- La raíz establece el bit de TC durante un período igual a la suma de la antigüedad máxima y el retardo de envío (en segundos), que de manera predeterminada es  $20+15=35$

## Notificación de broadcast

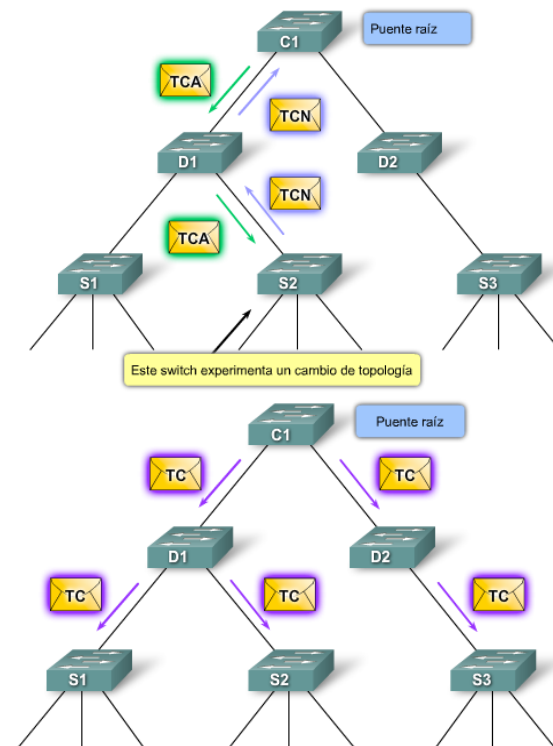
- Una vez que el puente raíz advierte que se ha producido un evento de cambio en la topología en la red, comienza a enviar sus BPDUs de configuración con el bit de cambio de topología (TC) establecido.
- Los switches reciben las BPDUs de cambio de topología tanto en los puertos en estado de enviar como en los de bloqueo.



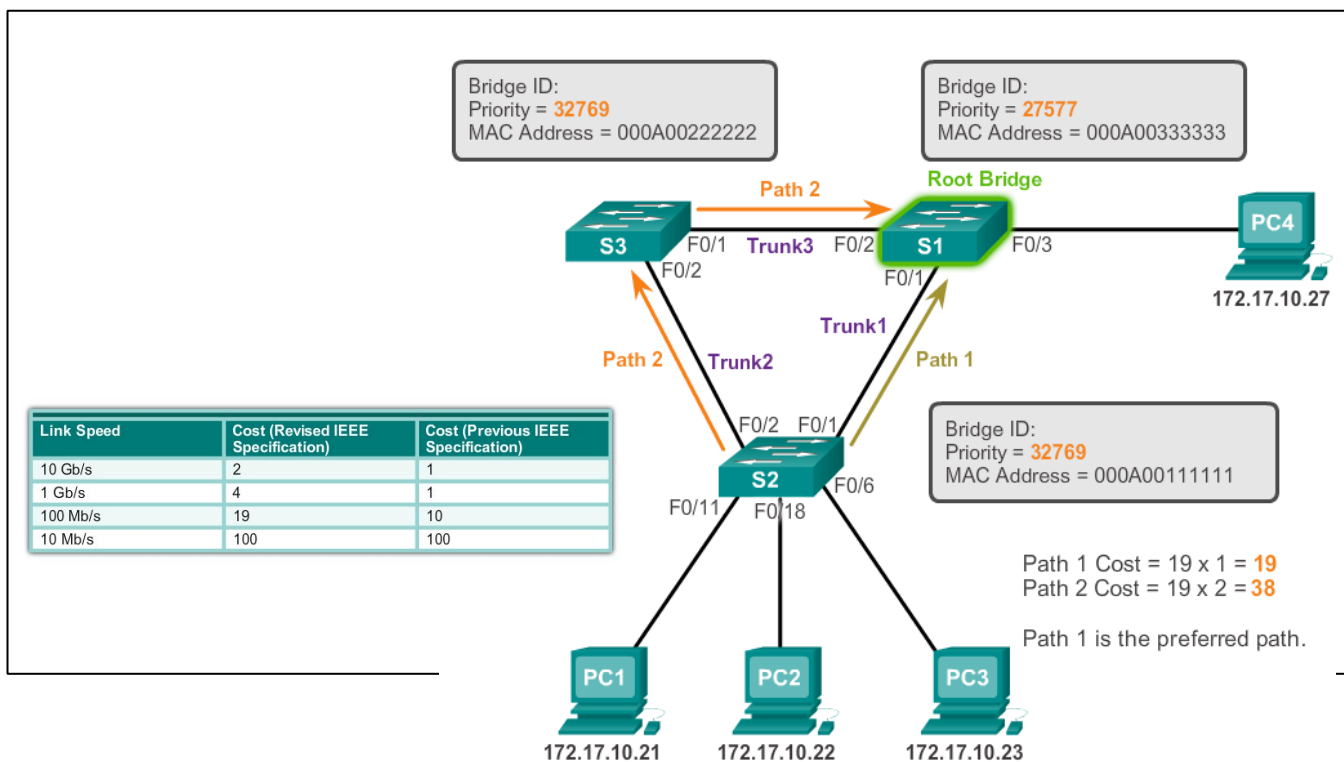
# Spanning-Tree: Notificación de cambio en la topología de STP

## Notificación de broadcast

- Una vez que el switch raíz advierte que se ha producido un cambio en la topología en la red, comienza a enviar sus BPDUs con el bit de cambio de topología (TC) establecido
- Los switches reciben las BPDUs de cambio de topología tanto en los puertos en estado de envío como en los de bloqueo
- La raíz establece el bit de TC durante un período igual a la suma de **máximo age** + **forward delay**, que de manera predeterminada es  $20+15=35$



# El Algoritmo Spanning-Tree: Coste de la Ruta

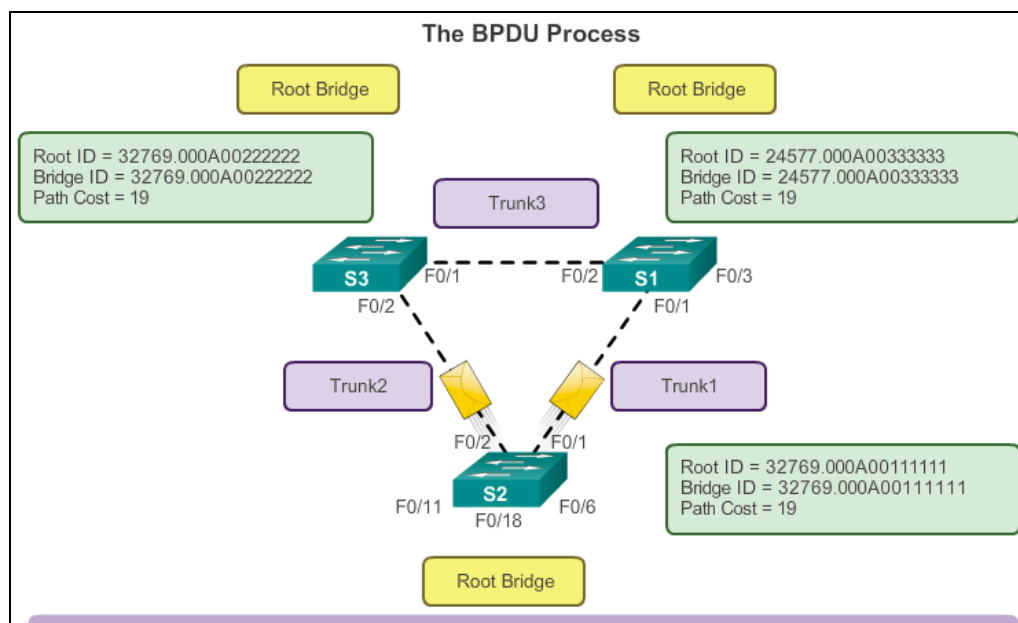


## Formato de Trama BPDU IEEE 802.1D

```
+ Frame 1 (60 bytes on wire, 60 bytes captured)
- IEEE 802.3 Ethernet
  + Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
  + Source: Cisco_9e:93:03 (00:19:aa:9e:93:03)
    Length: 38
    Trailer: 000000000000000000
  + Logical-Link Control
  - Spanning Tree Protocol
    Protocol Identifier: Spanning Tree Protocol (0x0000)
    Protocol version Identifier: Spanning Tree (0)
    BPDU Type: Configuration (0x00)
    + BPDU flags: 0x01 (Topology Change)
      Root Identifier: 24577 / 00:19:aa:9e:93:00
      Root Path Cost: 0
      Bridge Identifier: 24577 / 00:19:aa:9e:93:00
      Port identifier: 0x8003
      Message Age: 0
      Max Age: 20
      Hello Time: 2
      Forward Delay: 15
```

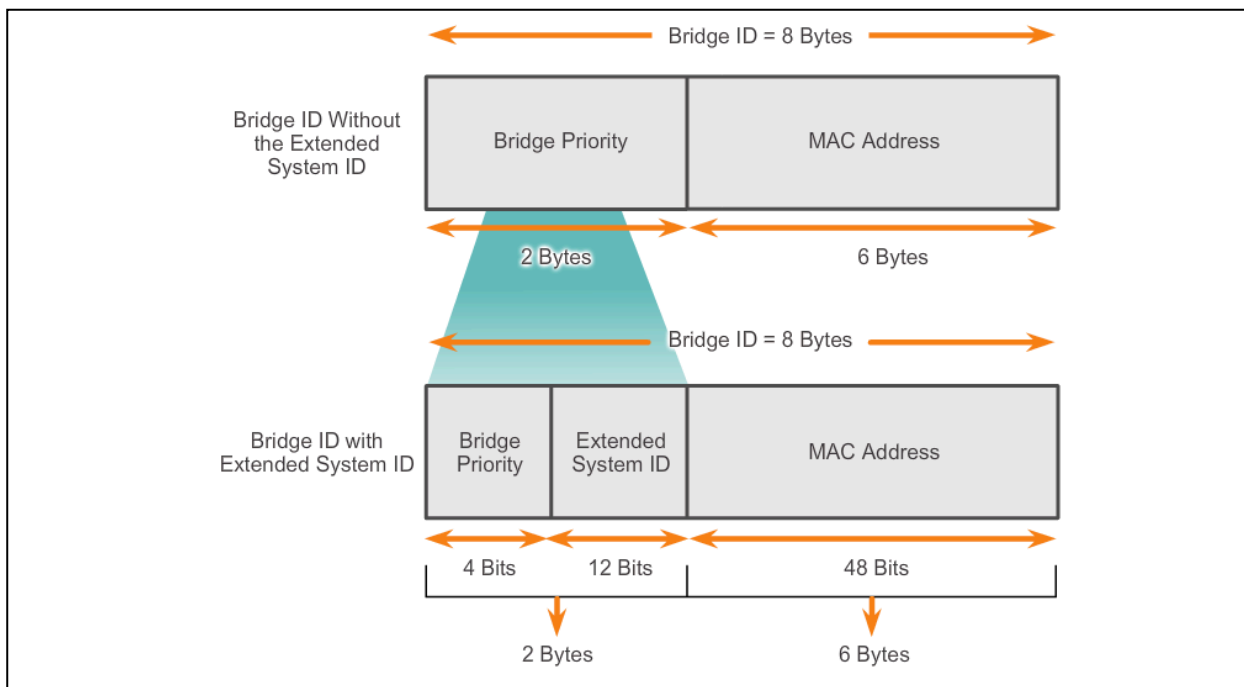
ESTO ES LO QUE CONTIENE UNA TRAMA BPDU

## Proceso y Propagación de BPDUs



El switch 2 envía BPDUs por todos sus puertos. Las BPDUs contienen el Bridge ID y el Root ID del Switch S2, indicando que él es el switch raíz

## ID de Sistema Extendido



## STP Portfast

- STP *portfast* deshabilita la generación de TCN BPDUs y provoca que los puertos de acceso (edge ports) se salten las fases de *listening* y *learning* y pasen directamente a *forwarding*
- Si se recibe una BPDU por dicho puerto (cosa que no debería suceder), la funcionalidad de *portfast* se elimina
- Comandos

Command	Description
<code>spanning-tree portfast</code>	Comando que se aplica en el modo de configuración de interfaces y que active el comportamiento <i>portfast</i> en un puerto de acceso específico
<code>spanning-tree portfast default</code>	Comando del modo configuración global en el que se aplica la configuración <i>portfast</i> a todos los puertos en modo acceso
<code>spanning-tree portfast disable</code>	Disable portfast on a port
<code>spanning-tree portfast trunk</code>	Command used on trunk links to enable portfast *This command should only be used with ports connected to a single host.

## STP Portfast

- STP *portfast* deshabilita la generación de TCN BPDUs y provoca que los puertos de en modo acceso se salten las fases de *listening* y *learning* y pasen directamente a *forwarding*
- Si se recibe una BPDU por dicho puerto (cosa que no debería suceder), la funcionalidad de *portfast* se elimina
- Comandos

Command	Description
<code>spanning-tree portfast</code>	Comando que se especifica en una interfaz y que sirve para activar este comportamiento en dicha interfaz. Se debe aplicar en puertos de acceso.
<code>spanning-tree portfast default</code>	Comando que se aplica en el modo de configuración global y habilita <i>portfast</i> en todos los puertos de acceso
<code>spanning-tree portfast disable</code>	Deshabilita <i>portfast</i> en una interfaz específica, si se ha habilitado globalmente
<code>spanning-tree portfast trunk</code>	Este commando se utiliza para activar <i>portfast</i> en los enlaces troncales. Este commando debe aplicarse únicamente a puertos conectados a un host



# Ejemplos de STP Portfast

- Ejemplos de configuración de Portfast

## Example 3-9 Enabling STP Portfast on Specific Interfaces

```
SW(config)# interface gigabitEthernet 1/0/13
SW(config-if)# switchport mode access
SW(config-if)# switchport access vlan 10
SW(config-if)# spanning-tree portfast
```

```
SW# show spanning-tree vlan 10
```

```
! Output omitted for brevity
```

```
VLAN0010
```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Gi1/0/2	Desg	FWD	4	128.2		P2p
Gi1/0/3	Desg	FWD	4	128.3		P2p
Gi1/0/13	Desg	FWD	4	128.13		P2p Edge

```
Gi1/0/2
```

```
Desg
```

```
FWD
```

```
4
```

```
128.2
```

```
P2p
```

```
Gi1/0/3
```

```
Desg
```

```
FWD
```

```
4
```

```
128.3
```

```
P2p
```

```
Gi1/0/13
```

```
Desg
```

```
FWD
```

```
4
```

```
128.13
```

```
P2p
```

```
Edge
```

```
SW# show spanning-tree interface gi1/0/13 detail
```

```
Port 13 (GigabitEthernet1/0/13) of VLAN0010 is designated forwarding
```

```
Port path cost 4, Port priority 128, Port Identifier 128.7.
```

```
Designated root has priority 32778, address 0062.ec9d.c500
```

```
Designated bridge has priority 32778, address 0062.ec9d.c500
```

```
Designated port id is 128.7, designated path cost 0
```

```
Timers: message age 0, forward delay 0, hold 0
```

```
Number of transitions to forwarding state: 1
```

```
The port is in the portfast mode
```

```
Link type is point-to-point by default
```

```
BPDUs: sent 23103, received 0
```

## Example 3-10 Enabling STP Portfast Globally

```
SW# conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

```
SW(config)# spanning-tree portfast default
```

```
Warning: this command enables portfast by default on all interfaces. You
should now disable portfast explicitly on switched ports leading to hubs,
switches and bridges as they may create temporary bridging loops.
```

```
SW(config)# interface gi1/0/8
```

```
SW(config-if)# spanning-tree portfast disable
```

# Ejemplos de STP Portfast

- Ejemplos de configuración de Portfast

## Example 3-9 Enabling STP Portfast on Specific Interfaces

```
SW(config)# interface gigabitEthernet 1/0/13
SW(config-if)# switchport mode access
SW(config-if)# switchport access vlan 10
SW(config-if)# spanning-tree portfast
```

```
SW# show spanning-tree vlan 10
```

```
! Output omitted for brevity
```

```
VLAN0010
```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Gi1/0/2	Desg	FWD	4	128.2		P2p
Gi1/0/3	Desg	FWD	4	128.3		P2p
Gi1/0/13	Desg	FWD	4	128.13		P2p Edge

```
Gi1/0/2
```

```
Gi1/0/3
```

```
Gi1/0/13
```

```
SW# show spanning-tree interface gi1/0/13 detail
```

```
Port 13 (GigabitEthernet1/0/13) of VLAN0010 is designated forwarding
```

```
Port path cost 4, Port priority 128, Port Identifier 128.7.
```

```
Designated root has priority 32778, address 0062.ec9d.c500
```

```
Designated bridge has priority 32778, address 0062.ec9d.c500
```

```
Designated port id is 128.7, designated path cost 0
```

```
Timers: message age 0, forward delay 0, hold 0
```

```
Number of transitions to forwarding state: 1
```

```
The port is in the portfast mode
```

```
Link type is point-to-point by default
```

```
BPDUs: sent 23103, received 0
```

## Example 3-10 Enabling STP Portfast Globally

```
SW# conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

```
SW(config)# spanning-tree portfast default
```

```
Warning: this command enables portfast by default on all interfaces. You
should now disable portfast explicitly on switched ports leading to hubs,
switches and bridges as they may create temporary bridging loops.
```

```
SW(config)# interface gi1/0/8
```

```
SW(config-if)# spanning-tree portfast disable
```

## Versiónes de Spanning Tree

- STP ha sido desarrollado en múltiples iteraciones:
  - 802.1D, que es la especificación original
  - Per-VLAN Spanning Tree (PVST)
  - Per-VLAN Spanning Tree Plus (PVST+)
  - 802.1W Per-VLAN Rapid Spanning Tree Protocol (PVRSTP)
  - 802.1S Multiple Spanning Tree Protocol (MST)
- Los switches Catalyst, en la actualidad, operan en PVST+, PVRSTP o MST
- Cualquiera de estos tres modos es compatible con 802.1D